10. import org.junit.jupiter.params.ParameterizedTest;

import org.junit.jupiter.params.converter.ConvertWith;

import org.junit.jupiter.params.provider.CsvSource;

import static org.junit.jupiter.api.Assertions.\*;

public class ArgumentConversionExample {

@ParameterizedTest

@CsvSource({"John, 25, true", "Jane, 30, false"})

public void testPerson(@ConvertWith(PersonConverter.class) Person person) {

assertNotNull(person);

assertTrue(person instanceof Person);

assertEquals(3, person.getAttributes().size());

assertEquals("name", person.getAttributes().get(0).getName());

assertEquals("age", person.getAttributes().get(1).getName());

assertEquals("isAdult", person.getAttributes().get(2).getName());

}

}

class Person {

private List<Attribute> attributes;

public Person(List<Attribute> attributes) {

this.attributes = attributes;

}

public List<Attribute> getAttributes() {

return attributes;

}

}

class Attribute {

private String name;

private Object value;

public Attribute(String name, Object value) {

this.name = name;

this.value = value;

}

public String getName() {

return name;

}

public Object getValue() {

return value;

}

}

class PersonConverter extends SimpleArgumentConverter {

@Override

protected Object convert(Object source, Class<?> targetType) throws ArgumentConversionException {

if (source instanceof String) {

String[] parts = ((String) source).split(",");

List<Attribute> attributes = new ArrayList<>();

attributes.add(new Attribute("name", parts[0]));

attributes.add(new Attribute("age", Integer.parseInt(parts[1])));

attributes.add(new Attribute("isAdult", Boolean.parseBoolean(parts[2])));

return new Person(attributes);

} else {

throw new IllegalArgumentException("Conversion failed: " + source);

}

}

}